Having thus described our invention, we now claim:

1.   1.   A method for administration and replication of a database, comprising the

2   steps of:

3   providing a database management system with a built-in random

4   sampling facility integrated into said database management system; and,

5   executing said random sampling facility from within the database

6   management system to perform a replication operation on said database.

1   2.   The method as set forth in claim 1, further comprising the steps of:

2   defining a database record sample size S;

3   randomly sampling S records of the database using said random sampling

4   facility;

5   storing statistics for each of said S records, wherein said statistics include

6   a record key for each record; and,

7   producing an extrapolated replication partition analysis based on said

8   statistics.

1   3.   The method as set forth in claim 2, wherein the step of defining said

2   sample size S includes:

3   defining a default sample size;

4   selectively receiving a desired sample size; and,

STL920000104US1

16

5   setting said sample size S as said default sample size when the desired

6   sample size is not selectively received, and setting said sample size S as said desired sample size

7   when the desired sample size is selectively received.

1   4.   The method as set forth in claim 1, further comprising the steps of:

2   defining a database record sample size S;

3   randomly sampling S records of the database using said random sampling

4   facility;

5   storing statistics for each of said S records, wherein said statistics include

6   a record key for each record; and,

7   producing a partial replication partition analysis based on said statistics.

1   5.   The method as set forth in claim 4, wherein the step of defining said

2   sample size S includes:

3   defining a default sample size;

4   selectively receiving a desired sample size; and,

5   setting said sample size S as said default sample size when the desired

6   sample size is not selectively received, and setting said sample size S as said desired sample size

7   when the desired sample size is selectively received.

1   6.   A method for database administration and replication, comprising the

2   steps of:

3   providing a database management system with an integrated random

4   sampling facility;

5   selecting a default sample size value S;

6   selectively receiving a desired sample size value D and setting said

7   default sample size value S to said desired sample size value D when said desired sample size

8   value D is received;

9   randomly sampling S records of the database using said random sampling

10   facility;

11   storing statistics for each of said S records, wherein said statistics include

12   a record key for each record; and,

13   producing at least one of:

14   an extrapolated replication partition analysis based on said

15   statistics; and

16   a partial replication partition analysis based on said statistics.

1   7.   The method as set forth in claim 6, wherein the step of selecting said

2   default sample size value D further includes the steps of:

3   generating a table of S number pairs (Y_j, I_j), j=1,2,...,S, wherein all Y and

4   all I are initially set to zero;

5   initializing a reservoir of records to an empty +state;

6   setting an index M to said reservoir equal to zero;

7   generating a sequence of N non-repeating random numbers $U_1, U_2, ..., U_N$,

8   $0 \leq U \leq 1$, wherein N is the number of records in the database; and,

9   performing additional steps for each random number U_k generated,

10   k=1,2,...,N, the additional steps including:

11   skipping the next record in the database if U_k is less than the

12   .   smallest value of Y in said table of number pairs; and,

13   updating the table if a Y less than U_k exists by performing

14   further steps including:

15   setting M equal to its current value plus one;

16   replacing the smallest Y in the table with U_k;

17   setting the I value paired with the smallest Y equal

18   to M; and,

19   storing all or part of the next record of the

20   database in said reservoir of stored records, wherein the current value of

21   M is a reservoir index to said stored record.

1   8.   The method as set forth in claim 7, wherein the step of updating the table

2   further includes the step of:

3   arranging the table in a heap with respect to Y.

1   9.   The method as set forth in claim 6, further comprising the step of:

2   sorting said stored statistics by key prior to producing said partition

3   analysis.

1   10.   The method as set forth in claim 9, further comprising the steps of:

2   accessing all database records in an arbitrary sequence;

            iteratively filling all of said partitions except the last said partition with said accessed records to a maximum byte count; and,

            storing remaining accessed records in the last of said partitions.

1   11.   The method as set forth in claim 6, wherein the step of storing statistics

2 includes storing said statistics in a memory.

1   12.   The method as set forth in claim 11, wherein the step of storing statistics

2 includes storing said statistics in said memory in a compressed format.

1   13.   The method as set forth in claim 6, wherein the step of producing at least

2 one of said partition analyses includes the step of defining multiple partition boundaries.

1   14.   The method as set forth in claim 6, wherein the step of sampling said S

2 records includes randomly sampling the S records utilizing dataspaces including:

3   at least one index dataspace;

4   at least one key dataspace; and,

5   at least one statistics dataspace.

1   15.   A database management system (DBMS) for managing an associated

2   database, the DBMS comprising:

3   random sampling facility integrated with the database management

4   system;

5   first database analysis tools using said integrated random sampling

6   facility for generating extrapolated reports on database content;

7   second database analysis tools using said integrated random sampling

8   facility for generating extrapolated reports on database size; and,

9   database replication tools adapted to execute at least one of a complete

10   replication having output partition sizes determined by extrapolating a random sample of said

11   database, and a partial replication in which the data stored in the partial replication comprises a

12   random sample of said database.

1   16.   The database management system of claim 15 further comprising:

2   a pre-configured number S defining a default sample size;

3   a means for selectively receiving a particular number defining a desired

4   sample size and setting said number S equal to said particular number;

5   a means for randomly sampling S records of the database using said

6   random sampling facility;

7   a means for storing statistics for each of said S records, wherein said

8   statistics include a record key for each record; and,

9   a means for producing at least one of:

10   an extrapolated database content analysis based on said statistics;

11   an extrapolated partition analysis based on said statistics; and,

12   a partial partition analysis based on said statistics.

1   17.   The database management system of claim 16, further comprising:

2   a means for sorting said stored statistics by key prior to producing at least

3 one of said analyses.

1   18.   The database management system of claim 16, wherein said means for

2 randomly sampling S records further comprises:

3   a means for generating a table of S number pairs (Y_j, I_j), $j=1,2,...,S$,

4 wherein all Y and all I are initially zero;

5   a means for initializing a reservoir of records to an empty state;

6   a means for setting an index M to said reservoir equal to zero;

7   a means for generating a sequence of N non-repeating random numbers

8 $U_1, U_2,..., U_N$, $0 \leq U \leq 1$, wherein N is the number of records in the database; and,

9   a means, for each random number U_k generated, $k=1,2,...,N$, comprising:

10   a means to skip the next record in said database if U_k is

11 less than the smallest value of Y in said table of number pairs; and,

12   a means to update the table if a Y less than U_k exists,

13 comprising:

14   a means to set M equal to its current value plus one;

15   a means to replace the smallest Y in the table with U_k;

16   a means to set the I value paired with the smallest Y equal

17   to M; and,

18   a means to store all or part of the next record of said

19   database in said reservoir of stored records, wherein the current value of

20   M is a reservoir index to said stored record.

1   19.   The database management system of claim 18 wherein the means to

2 update the table further comprises:

3   a means to arrange the table in a heap with respect to Y.

1   20.   The database management system of claim 18, wherein said means for

2 storing statistics comprises a means for storing said statistics in memory.

1   21.   The database management system of claim 20, further comprising a

2 means for sorting said stored statistics by key prior to producing at least one of said analyses.

1   22.   The database management system of claim 21, wherein said partition

2 analyses include analyses of multiple partition boundaries.

1   23.   The database management system of claim 22, further comprising:

2   a means for accessing all database records in an arbitrary sequence;

3   a means for iteratively filling all of said partitions except the last with said

4 accessed records to a maximum byte count; and,

5   a means for storing remaining accessed records in the last of said

6 partitions.

1   24. The database management system of claim 16, further comprising:

2   a means for utilizing at least one index dataspace;

3   a means for utilizing at least one key dataspace; and,

4   a means for utilizing at least one statistics dataspace.